

# Relatório do Trabalho Prático

## Projeto Integrado



Ano Letivo 2019/2020

1º Semestre

Docentes: Cristóvão Sousa e João Ribeiro

Pedro Mendes – 8180666

Carlos Guimarães - 8160321

## Índice

Índice de Figuras .....	3
Introdução .....	4
Desenvolvimento .....	5
O Projeto .....	5
Mockups .....	5
Montagem do robô .....	7
Instalação do Sistema Operativo no robô .....	8
Conexão do robô com o computador .....	8
Descrição do projeto .....	9
Main Activity .....	11
Programação do Robô .....	12
Detetar cores .....	14
Conclusão .....	15

## Índice de Figuras

Figura 1- Mockup inicial.....	6
Figura 2- Mockup modo jogo .....	6
Figura 3 - Mockup menu.....	7
Figura 4- Montagem .....	7
Figura 5 - Lejos.....	8
Figura 6 - Registo .....	9
Figura 7 - Login .....	9
Figura 8 - Recuperação de conta .....	10
Figura 9 - Firebase .....	10
Figura 10 - Histórico de movimentos .....	11
Figura 11 - Main Activity.....	11
Figura 12- Modo Jogo .....	12
Figura 13 - Excerto de código 1 .....	13
Figura 14 - Excerto de código 2 .....	13
Figura 15 - Robô a detetar cores .....	14

## Introdução

O presente projeto foi elaborado no âmbito da disciplina de Projeto Integrado do Curso Técnico Superior de Desenvolvimento para a Web e Dispositivos Móveis.

O projeto consiste numa aplicação android para controlar um robô da marca lego, o Mindstorm EV3.

Para a elaboração do projeto foi desenvolvida uma aplicação que permite movimentar o robô em todas as direções, movimentos autónomos do robô e deteção de algumas cores por parte do mesmo. No último ponto, para mostrar o resultado que detetou escreve no ecrã e coloca as suas luzes com a cor correspondente.

Neste relatório o projeto realizado é descrito pormenorizadamente, sendo abordada a forma como foi elaborado, o modo de funcionamento e todas as questões tidas em atenção.

## Desenvolvimento

### O Projeto

O projeto elaborado é uma aplicação android capaz de controlar um robô por Bluetooth e detetar cores.

Para o projeto ser possível, várias questões foram tidas em atenção.

Foi necessário instalar no robô o sistema operativo Lejos e alterar algumas definições no computador para os dois poderem comunicar via cabo USB. Depois de estar efetuada a ligação entre o computador e o robô foi necessário criar uma aplicação com o Android Studio para comunicar com o robô remotamente. De seguida, procedeu-se à programação do robô para o conectar por Bluetooth e para receber informações em bytes para serem tratados posteriormente. A partir do momento que o robô começou a receber bytes foi necessário tratar os dados, ou seja, o robô recebe a informação em bytes e teve de se manipular o robô para que fizesse o que era pretendido.

Depois do robô já estar a movimentar-se, foi instalado o sensor de cores e implementou-se um novo programa para fazer todos os testes necessários.

### Mockups

Quando se inicia um novo projeto, um passo extremamente útil é elaborar os mockups da aplicação. Dessa forma coloca-se detalhadamente tudo que é pretendido ter na aplicação e fica simples de seguir um caminho específico.

Inicialmente foram elaborados mockups, no entanto no decorrer da elaboração do trabalho foi possível perceber que os mockups produzidos não correspondiam totalmente ao objetivo do projeto. Devido ao referido anteriormente, o produto final não corresponde na totalidade ao implementado nos mockups.

### Página inicial

Inicialmente a página inicial só iria conter um botão para fazer a conexão com o robô.

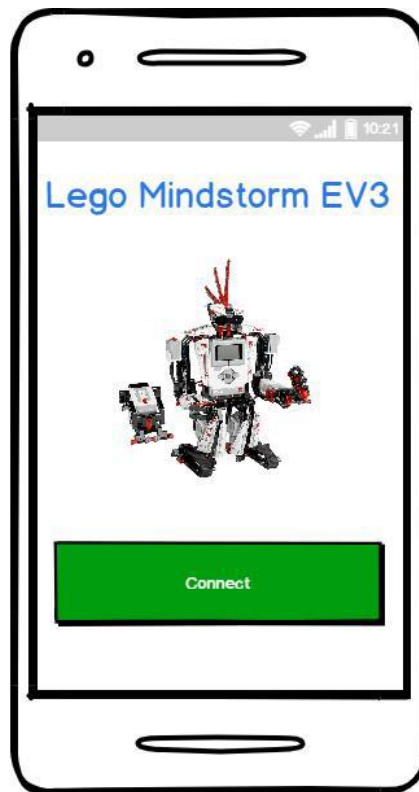


Figura 1- Mockup inicial

### Modo Jogo

O modo jogo teria do lado direito da APP os botões de navegação do robô e na parte esquerda da APP os botões para fazer algumas atividades como por exemplo andar sozinho e detetar cores.

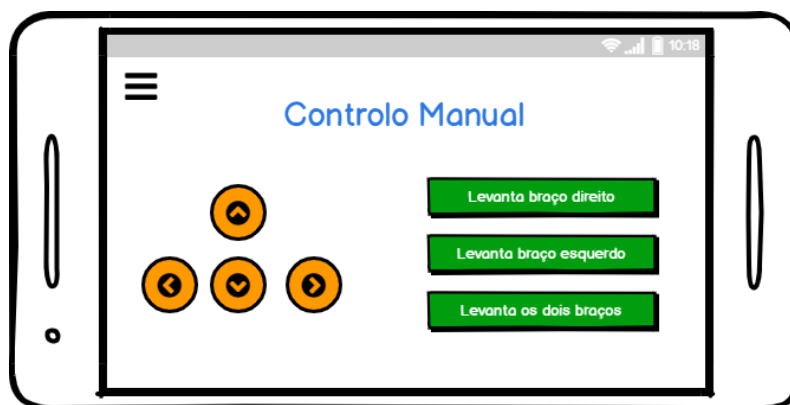


Figura 2- Mockup modo jogo

## Menu

Nos mockups pensava-se que seria de fácil utilização o utilizador poder aceder ao menu através de todas as páginas.

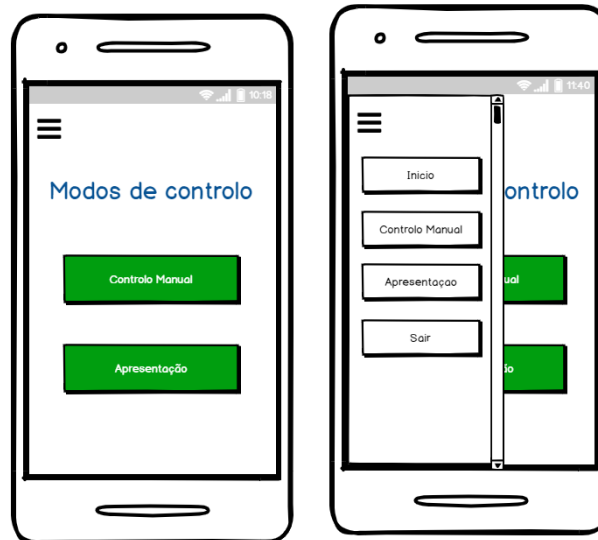


Figura 3 - Mockup menu

## Montagem do robô

O Mindstorm EV3 conta com 541 peças, pelo que é um processo complexo e demorado. Seguem-se algumas imagens da montagem.

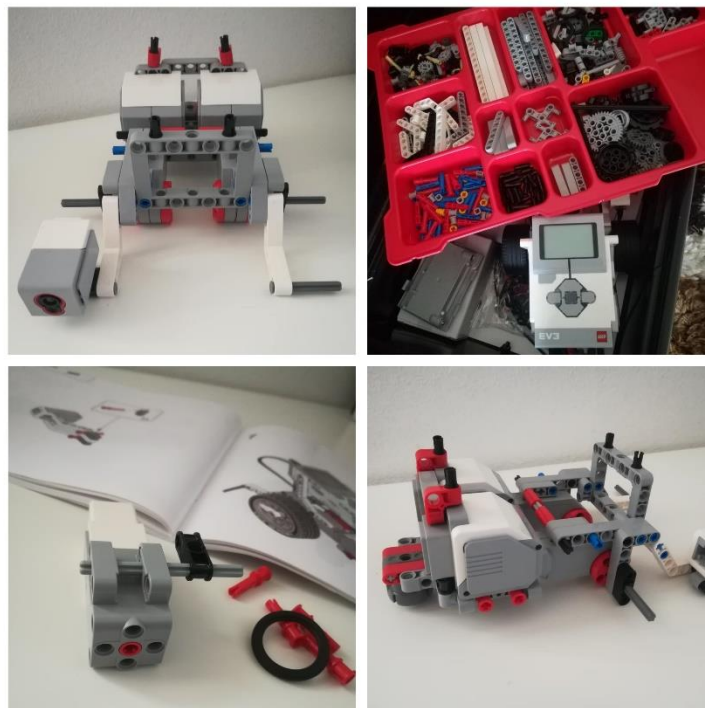
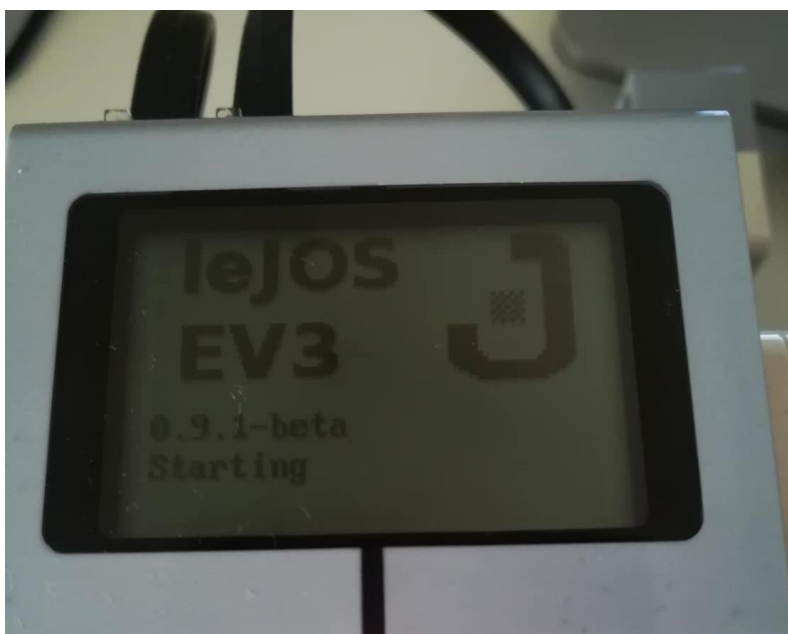


Figura 4- Montagem

### Instalação do Sistema Operativo no robô

Para a instalação do sistema operativo foi utilizado um cartão de memória de 32GB e repartiu-se o cartão de memória para o Lejos poder ser instalado. Posteriormente foi instalado um programa específico no computador e procedeu-se ao download do Lejos.

O programa instalou o Lejos no cartão de memória e após esse passo apenas foi necessário implementar o cartão no robô.



*Figura 5 - Lejos*

### Conexão do robô com o computador

Na conexão do robô com o computador foi onde surgiram os maiores problemas.

Para fazer a conexão vários passos precisam ser seguidos, principalmente num computador com sistema operativo Windows instalado. É precisa uma versão do java especifica no Lejos e uma outra especifica no computador. Foi também necessário instalar um driver para reconhecer o robô não como porta USB, mas como dispositivo RNDIS. Depois de já estar a reconhecer o robô no computador foi necessário configurar uma porta USB e partilhar com a rede em que se está ligado.



## Descrição do projeto

### *Login, Registo*

Na figura 6 é possível observar o formulário de registo na página.

Caso seja a primeira vez do utilizador na aplicação, este deve efetuar os passos necessários do formulário de registo. Após o registo, este poderá iniciar sessão e começar a utilizar o robô.

Caso já tenha conta criada e apenas deseja aceder à sua conta, este poderá efetuar o login como indica na figura 7.

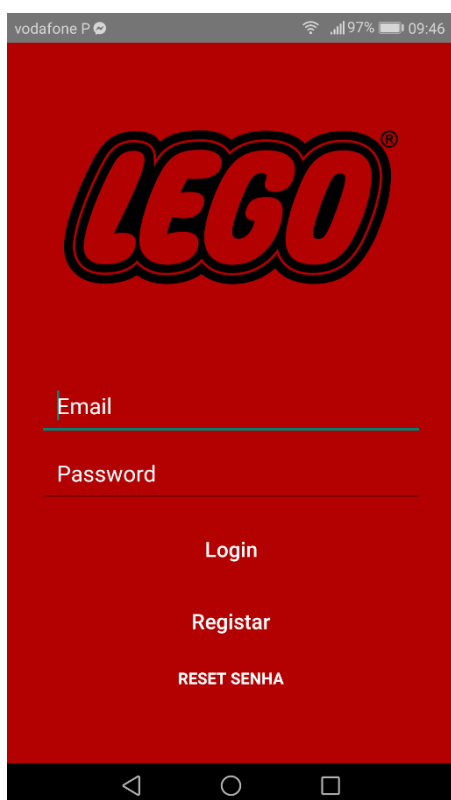


Figura 7 - Login

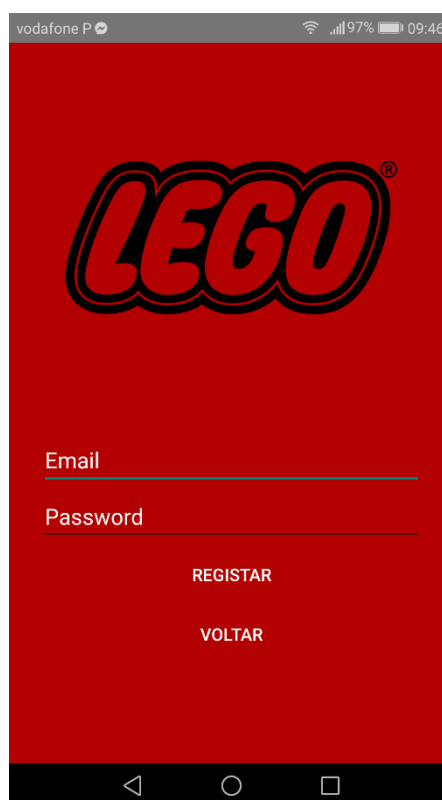


Figura 6 - Registo

### *Recuperação de Conta*

Na figura 8 é possível observar o formulário de recuperação de conta na aplicação.

Este formulário permite fazer a recuperação da palavra passe, caso o utilizador se tenha esquecido.

Através de um email será enviado um link onde poderá renovar a palavra-passe de acesso.

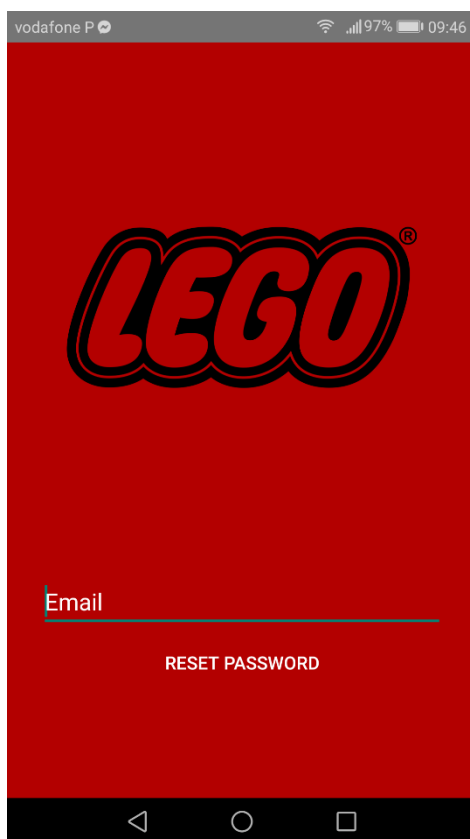


Figura 8 - Recuperação de conta

## Firestore

Ao realizar o registo, a conta do utilizador ficará guardada na base de dados do Firestore como mostra a figura 9.

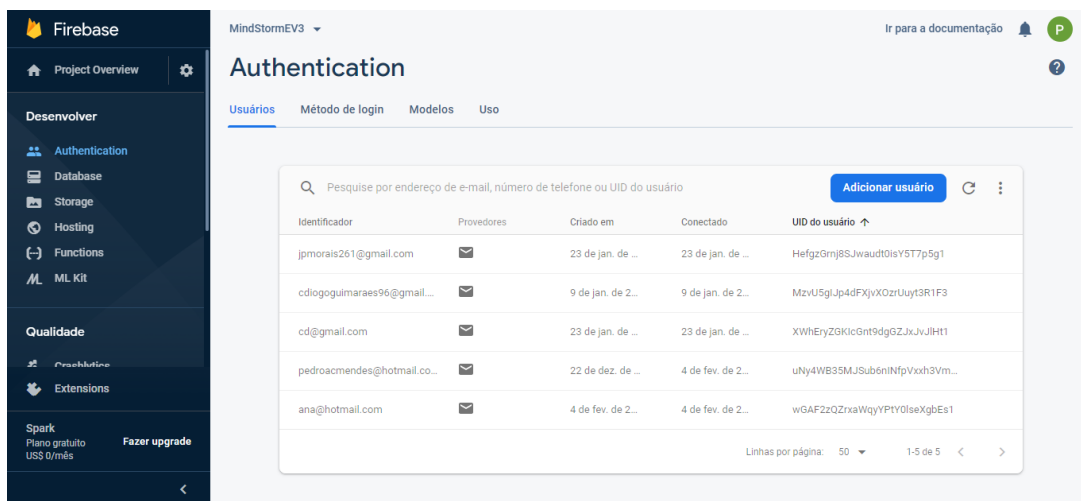


Figura 9 - Firestore

No entanto também regista no firebase o histórico de movimentos e a data do movimento do robô, como demonstra a figura 10.

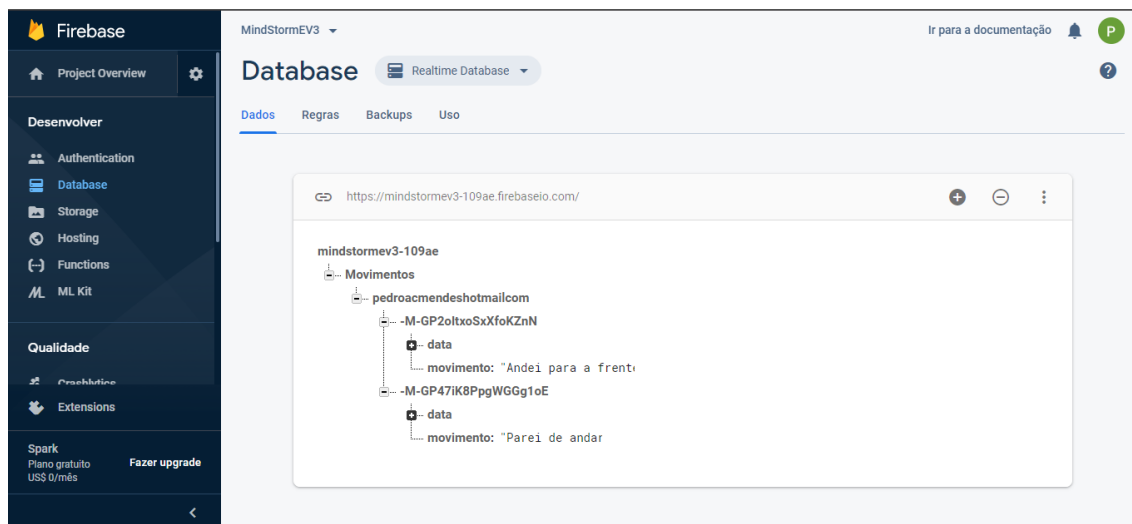


Figura 10 - Histórico de movimentos

## Main Activity

Na figura seguinte encontra-se a página onde o utilizador pode abrir o modo de jogo ou então sair da aplicação. Sempre que entrar nesta página será ligado o Bluetooth do telemóvel caso este esteja desligado, caso contrário não poderá aceder a página seguinte.



Figura 11 - Main Activity

### Modo Jogo

Na figura seguinte encontra-se a página onde o utilizador pode controlar o robô por Bluetooth, tendo acesso a vários botões de controlo que são capazes de fazer o robô andar nas direções que pretende, fazer o robô andar sozinho e detetar cores.



Figura 12- Modo Jogo

### Programação do Robô

A programação para o robô consiste em 3 classes, sendo que a parte que o faz movimentar é relativamente pequena e simples (figura 14).

Relativamente à parte de enviar e receber a informação foi onde surgiram os maiores problemas. A figura 13 demonstra um excerto do código relativo à conexão Bluetooth.

```
static void startBluetoothConnection() {
    NXTConnection connection = null;
    try {
        System.out.println("init BT connection ...");
        connection = Bluetooth.getNXTCommConnector().waitForConnection(10000, NXTConnection.RAW);
        inputStream = connection.openDataInputStream();
        outputStream = connection.openDataOutputStream();
        System.out.println("Connected");
    }
    catch (Exception a) {
        System.out.println("ERROR");
        Bluetooth.getNXTCommConnector().cancel();

        try {
            inputStream.close();
            outputStream.close();
            connection.close();
        } catch (IOException e) {
            System.out.println("ERROR: closing failed");
        }
        startBluetoothConnection();
    }
    getData();
}
```

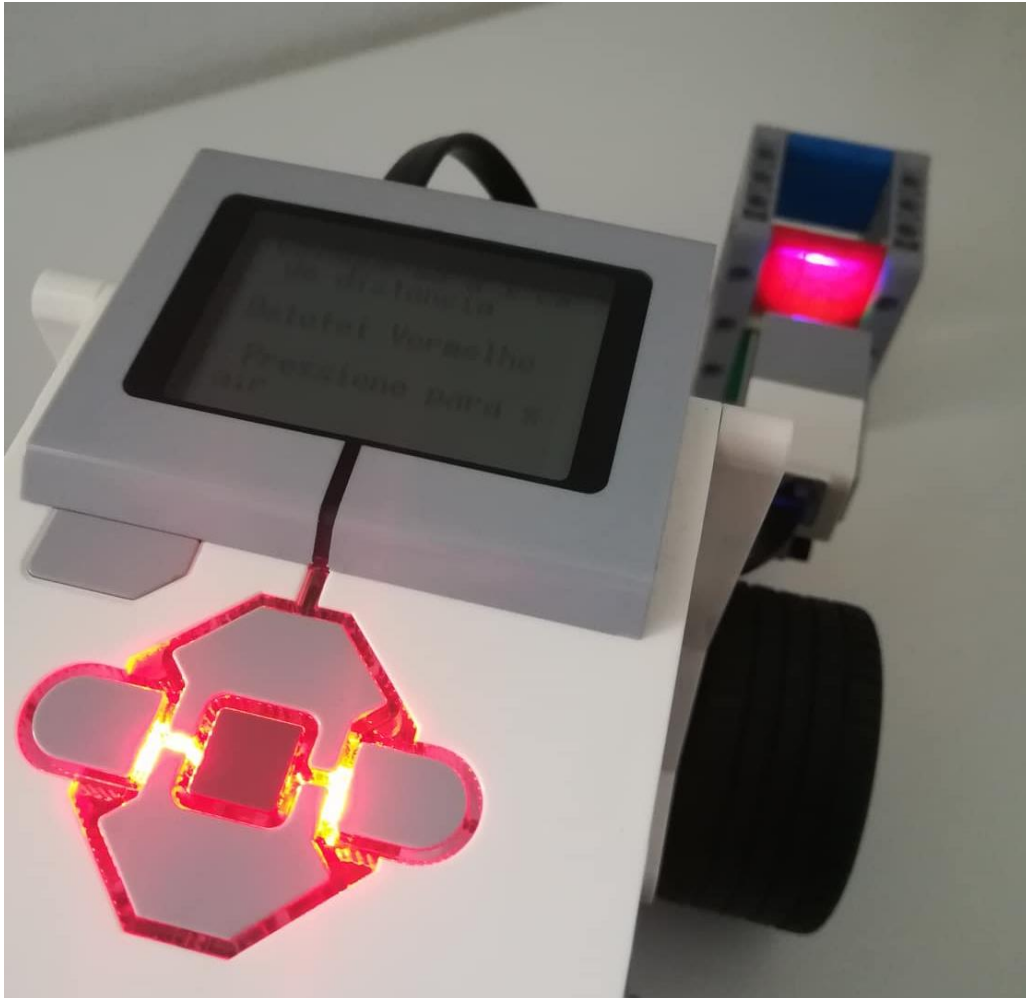
Figura 13 - Excerto de código 1

```
if (command == 943208504) {
    System.out.println("anda pra frente");
    leftMotor.setPower(50);
    rightMotor.setPower(50);
} else if (command == 825241904) {
    System.out.println("anda pra a direita");
    leftMotor.setPower(0);
    rightMotor.setPower(50);
} else if (command == 825504052) {
    System.out.println("anda pra a esquerda");
    leftMotor.setPower(50);
    rightMotor.setPower(0);
} else if (command == 842281524) {
```

Figura 14 - Excerto de código 2

## Detetar cores

A última parte do projeto foi proceder à deteção de cores pelo robô. Para isso, o robô necessita de estar próximo da cor que se pretende detetar, sendo o aconselhado entre 1 e 3 centímetros de distância para detetar corretamente a cor.



*Figura 15 - Robô a detetar cores*

## Conclusão

Através da elaboração desta aplicação foi possível adquirir mais conhecimentos sobre java, dispositivos móveis e também sobre robótica.

O projeto foi desenvolvido de uma forma simples e objetiva utilizando os recursos que nos foram fornecidos nas aulas. Para ter mais conhecimento acerca do robô e tudo que lhe está inerente foram feitas pesquisas na internet

Após as pesquisas e alguns dias de trabalho foi possível perceber que apesar de complicado seria um projeto exequível e interessante de desenvolver.

Durante a realização do trabalho foram sentidas dificuldades mais significativas com a conexão do robô e o computador. No entanto também foram sentidas outras dificuldades nomeadamente, ao pressionar um botão da aplicação para o robô se movimentar este só o fazia se fossem pressionadas duas vezes seguidas a mesma tecla. Para resolver este problema foi passada duas vezes a mesma informação em cada botão.

Desta forma o trabalho serviu para ganhar prática no desenvolvimento de projetos dentro desta área que nos poderá servir de base para projetos maiores.